Effective Online Controlled Experiment Analysis at Large Scale

Aleksander Fabijan
Malmö University
Dep. of Computer Science
Malmö, Sweden
aleksander.fabijan
@mau.se

Pavel Dmitriev
Microsoft, Analysis & Experimentation
Redmond, USA
padmitri
@microsoft.com

Helena Holmström Olsson
Malmö University
Dep. of Computer Science
Malmö, Sweden
helena.holmstrom.olsson
@mau.se

Jan Bosch
Chalmers University of Tech.
Dep. of Computer Science
Göteborg, Sweden
jan.bosch
@chalmers.se

***Abstract*—Online Controlled Experiments (OCEs) are the norm in data-driven software companies because of the benefits they provide for building and deploying software. Product teams experiment to accurately learn whether the changes that they do to their products (e.g. adding new features) cause any impact (e.g. customers use them more frequently). Experiments also help reduce the risk from deploying software by minimizing the magnitude and duration of harm caused by software bugs, allowing software to be shipped more frequently. To make informed decisions in product development, experiment analysis needs to be granular with a large number of metrics over heterogeneous devices and audiences. Discovering experiment insights by hand, however, can be cumbersome. In this paper, and based on case study research at a large-scale software development company with a long tradition of experimentation, we (1) describe the standard process of experiment analysis, and (2) introduce an artifact to improve the effectiveness and comprehensiveness of this process.**

Keywords— 'Online Controlled Experiments', 'A/B testing', 'Guided Experiment Analysis'

I. Introduction

Companies are successful only if they truly understand their customers' needs, and develop products or services that fulfil them [1]–[5]. Accurately learning about customer needs, however, is challenging and has always been an important part of software product development. Although asking customers through interviews, focus groups or other similar approaches remains essential for product development, product usage data such as telemetry or product logs [6], [7] enable software companies to become more accurate in evaluating whether their ideas add value to customers [3], [8], [9]. One of the enablers of this evolution is the internet connectivity of software products, which provides an unprecedented opportunity to evaluate ideas with customers in near real-time, and creates potential for making fast and accurate causal conclusions between the changes made to the product and the customers' reactions on them [10]–[13]. One way in which causal conclusions (for example, introducing a new feature causes customers to use the product more frequently) can be made is through A/B tests, or more generally, Online Controlled Experiments (OCEs) [3], [12], [14]. Experiments reduce the risk from deploying software by minimizing the magnitude and duration of harm caused by software bugs, set directions and goals for product teams, help save infrastructure resources, and deliver many other benefits to data-driven companies [14], [15]. Analyzing experiments at large scale, however, is challenging [16], [17]. In this paper, and based on a case study at Microsoft where over ten thousand experiments per year are conducted, we (1) present their current process and approach for analyzing OCEs, and (2) develop an artifact that improves the analysis process.

II. Background

Experimentation in software product development is an increasingly active research area [2], [3], [5], [9], [10], [18]–[21]. The theory of controlled experiments itself, however, dates to Sir Ronald A. Fisher's experiments at the Rothamsted Agricultural Experimental Station in England during the 1920s [22]. In the simplest controlled experiment, two comparable groups are created by randomly assigning experiment participants in either of them; the control and the treatment. The only thing different between the two groups is a change X. For example, if the two variants are software products, they might have different design solutions or communicate with a different server. If the experiment were designed and executed correctly, the only thing consistently different between the two variants is the change X. External factors such as seasonality, impact of other product changes, competitor moves, etc. are distributed evenly between control and treatment. Hence any difference in metrics between the two groups must be due to the change X (or a random chance, that is ruled out using statistical testing). This design establishes a causal relationship between the change X made to the product and changes in user behavior, measured through metrics.

For effective experimentation, a product team should identify and formalize a set of metrics that should describe long-term goals of the product. As the number of metrics grows, companies organize their metrics in semantically meaningful groups. Conceptually, the following four groups of metrics have been recognized as useful for analyzing experiments in previous research [17], [23], [24]: *Success Metrics* (the metrics that feature teams should improve), *Guardrail Metrics* (the metrics that are constrained to a band and should not move outside of that band), *Data Quality Metrics* (the metrics that ensure that the experiments are set-up correctly, and that no quality issues happened), and *Debug Metrics* (the drill down into success and guardrail metrics).

III. Research Method

This case study research [25] builds on an ongoing work with Microsoft. A single OCE at Microsoft can be started in a few minutes, and most experiments have hundreds of thousands to millions of users in their analysis.

A. Data Collection

The study is based on data collected at the case company during April 2016 and August 2017. It consists of meeting notes, interviews, observations, documentation, and prototype evaluations. The first author of this paper has been collaborating with the case company for over two years. During this time, we collected historical data about experiment analysis, participated and took notes at daily meetings where recent experiment analyses were discussed, interviewed case company participants, and conducted evaluation sessions. The second author of this paper is employed at the case company for over seven years and worked closely with many product groups to help them conduct OCEs.

B. Data Analysis

To discover how the experiment analysis process is conducted at the case company, we analyzed our empirical data using thematic coding approach, and triangulated the notes from experiment analysis observations with historical OCEs data and interview notes. Specifically, we were interested in identifying the most common analysis steps used in this process.

IV. Analyzing Online Controlled Experiments

In this section, we present the analysis of online controlled experiments at our case company.

A. Standard Scorecard Analysis

At our case company, experiments are analyzed through scorecards [26]. In a typical scorecard, the most relevant information about experiment metric movements is displayed. If we take an experiment with one control and one treatment as an example, the scorecard displays, for every metric, the result of the metric for the control and the treatment (for example, an average of clicks), the difference between the two (sometimes expressed as a percentage of change relative to the control - %delta), and most importantly, the p-value, denoting the level of statistical significance. If effect size is large, it can be informative to show it as well, however, %delta typically provides more information.

To make informed decisions, our case company organizes metrics in a scorecard in a semantically meaningful breakdown. For example, (see Figure 1), understanding why a success metric 'Overall Clicks Rate' has negatively reacted in the treatment group is easier to interpret with the additional knowledge that it was caused by a decrease of clicks on 'Web Results', while other components of the page had click increases. Furthermore, metrics are also grouped in similar conceptual groups as described in Section 2.

During the analysis process, every experiment is being examined for **(1) data quality issues, (2) decision making**, and **(3) deep-dive insight discovery** through segment-scorecards similar to the one on Figure 1.

Figure 1. Example metrics between a treatment (T) and a control (C).

Metric	T	C	Delta (%)	*p*-value
Overall Cllick Rate	0.9206	0.9219	-0.14%	8e-11
Web Results	0.5743	0.5800	-0.98%	~0
Answers	0.1913	0.1901	+0.63%	5e-24
Image	0.0262	0.0261	+0.38%	0.1112
News	0.0190	0.0190	+0.10%	0.8244

Data quality: Data quality is critical for trustworthy analysis and is the first step in experiment result analysis. Controlled experimentation allows for detection of very small changes in metrics that may not be detectable by monitoring the metric over time in a non-controlled setting. Thus, experiment analysis is very sensitive to data quality issues, even those which may not appear big in aggregate. One of the most effective data quality checks for experiment analysis is the Sample Ratio Mismatch (SRM) test, which utilizes the Chi-Squared Test to compare the ratio of the observed user counts in the variants against the configured ratio. When an SRM is detected, the results are deemed invalid and the experimenter is blocked from viewing the scorecard results (in order to prevent false conclusion making). While the SRM check applies to all types of products, every product or feature also contains data quality metrics tailored for their unique purpose. For example, a scorecard for an online page may contain metrics such as "number of JavaScript errors" and "page load time", while a scorecard for a mobile app contains metrics such as "telemetry loss rate" or "count of app crashes".

Decision Making: Product teams experiment with their design decisions, relevance modifications, infrastructure changes and other types of features in order to conduct a decision. Ideally, as companies mature, scorecards should clearly mark the subset of key metrics that the product organization agreed on being optimized for. Commonly, these are known as OEC (Overall Evaluation Criteria) and extensive research exists on designing key success metrics for decision making [3], [14], [17], [27]–[30]. A good OEC consists of success metrics - those that the experimenters are aiming to improve and which lead to the long-term desired outcomes for the company, and guardrails – metrics that express business constraints that the company does not want to degrade. For example, for a web site a success metric can be "number of visits per user", and a guardrail metric may be "ad revenue" (note that "ad revenue" is not a good success metric since increasing it via, e.g., showing users more ads may lead to users abandoning the site resulting in decrease in revenue over a long term [31]). The decision-making analysis is concerned with identifying how the success and guardrail metrics that the experimenter expected to impact in an experiment *actually* reacted to each treatment. E.g. if an experiment changed the appearance of an item on a web page with the intent to increase the engagement of the customers visiting the site, has this in fact resulted in the expected outcome? To get an initial understanding, experimenters analyze individual success and guardrail metrics. Although highlighting the metrics that actually changed in an experiment accelerates the analysis process, understanding the impact on success and guardrail metrics over dozens to hundreds of segments becomes a challenge.

In-Depth Analysis: Furthermore, in order to make informed decisions, experiments are typically examined beyond the aggregate analysis. At our case company, it is common for experiment scorecards to compute thousands of metrics over dozens of population segments, making manual scorecard analysis tedious and error prone. Important insights can be missed, especially if they are not expected by the experiment owner or heterogeneity is present in the treatment effect [16]. For example, product teams frequently segment their customer base in various ways in order to discover as many insights as possible to base their decisions on. Customers of a web application can be segmented by location, device type, gender, data center that they communicate with, to name a few. In practice, a scorecard similar to the one on Figure 1 needs to be reviewed for every segment, providing experimenters information specific to the segment (e.g. segment size, metric behavior, statistical significance level). At large scale where many experiments need to be analyzed, this poses a challenge in trustworthiness of decision making.

B. *Guided Experiment Analysis*

In analyzing a typical experiment, tens of different segment slices could be reviewed. This is a cumbersome and error prone approach. In this section, we demonstrate a possibility for a more effective experiment analysis using descriptive visualization for guiding experimenters to the most critical or interesting experiment scorecards - Guided Experiment Analysis (GEA).

In our artifact (a mockup of it is visible on Figure 2), experimenter is presented with a graphical illustration of an experiment at a segment level for every pair of experiment variations. The artifact consists of three columns. The first column depicts data quality behavior of the experiment, the second column contains the OEC behavior, and the third column depicts Segments of Interest (SOI) behavior. The number of rows is dynamic and depends on the available data collected from product telemetry. Every row represents a segmentation criterion which splits the experiment population into multiple segments, for which experimenters at the case company examine individual scorecards. For example, the segment 'Data Center' is divided into slice '1' for the customers connected to the first data center, '2' for those with the second, etc. The "Gender" segment is also further segmented into "Male", "Female" and "Unknown" slices, and so on with other segments. In the example from Figure 2, analysis is available for all of the product segments, on top of the "Aggregate" segment. An additional data point in this view is the **size of the cell**, which is proportional to the size of the segment slice. For example, experimenters using our artifact quickly see that the Gender segment slices in this experiment had a similar number of users.

Our artifact operates on sets of metrics. From the experimenter perspective, the artifact reveals whether there has been a significant change or not within a certain slice. At the same time, the experimenter has the ability to examine the slice in greater detail to identify which of the metrics contributed to the decisions by clicking on the individual slice and opening the Standard Scorecard for it. In this way, experimenters are guided from a 'bird's-eye-view' perspective to the segment slices where the likelihood of finding relevant experiment findings is high. What differs between the columns, and the critical information that leads to an accelerated discovery of experimental insights, is how the cells are colored.

Figure 2. Guided Experiment Analysis Approach.

1) Data Quality View

Experimenters analyze OCEs first for trustworthiness and data quality issues. Therefore, to provide trustworthy experiment analysis, the first column in our artifact highlights data quality issues on a segment slice level. Our artifact colors a cell in the data quality view green when no statistical difference in data quality metrics has been detected. In contrast, when either of the data quality metrics has a statistically significant change above or below a certain threshold (for example, the "telemetry loss rate" increases or decreases), the cell is colored red. In a scenario of a SRM, the cell with its counterpart in the OEC view is disabled and colored black, which can be seen on Figure 2 for the second data center.

2) OEC View

The second column in our artifact summarizes the key success and guardrail metric movement. When no guardrail or success metric movement is present, the cells are colored gray. This informs experimenters that the change in the experiment did not have an impact on that segment slice. When an experiment treatment only positively impacts the success metrics and no negative movement in the guardrail metrics is detected, the segment slice cell is colored green. When there is only negative movement in success or guardrail metrics, the cell color is red. Finally, it is possible for the results to be mixed (e.g. success metrics improve but guardrail metrics degrade), in which case the cell is colored purple, indicating that the tool cannot automatically determine whether this is good or bad. On example from Figure 2, experimenters learn that the "Aggregate" segment is positive, and that it was the male customers that contributed to this, whereas the female in fact reacted negatively. In such scenario, product teams can quickly start to form hypothesis about the possible causes for such heterogeneous movements, and instantly expand the understanding of the impact of their product changes.

3) Segments of Interest View

The third column in our Guided Experiment Analysis is the SOI view. Here, every segment slice is evaluated with an algorithm (developed prior to this research and also available in the SSA), and a score between 0 and 1 is assigned that depicts how

unusual its metric movements are. Our view colors the cells based on this score. Darker shades of blue indicate segment slices with highly unusual metric activity. In addition, the cells in this view display a number, indicating the count of metrics that have the activity recognized by the algorithm.

V. Conclusion and Future Work

Interpreting the results of OCEs is challenging even for the most experienced in the field [16], [17]. Experiment insights can easily be missed in the vast sets of segments and metrics, data-quality issues undetected, and the method of online controlled experimentation can quickly become untrustworthy. Efficient experiment analysis is thus critical for successful data-driven product development.

In this paper, we presented the analysis of online controlled experiments through Standard Scorecard Analysis, and how this can be improved through the proposed Guided Experiment Analysis approach. We believe that other software companies that conduct OCEs at scale can achieve similar benefits by incorporating Guided Experiment Analysis. For those that are just starting, however, the Standard Scorecard Analysis might be helpful and informative. And as the software industry is shifting into becoming increasingly more data-driven [20], experimentation practices are expected to be expanding to many domains, and growing where it is already present.

In future work, we aim to expand on the artifact with other angles of improving the OCEs analysis process, and carefully evaluate our work. Our findings suggest that the analysis of online controlled experiments remains an underexplored area from a software engineering perspective, and that companies can experience significant benefits from perceptually small improvements to their experimentation process.

References

[1] B. Boehm, "Value-based software engineering: reinventing," *SIGSOFT Softw. Eng. Notes*, vol. 28, no. 2, p. 3–, 2003.

[2] O. Rissanen and J. Munch, "Continuous Experimentation in the B2B Domain: A Case Study," *Proceedings - 2nd International Workshop on Rapid Continuous Software Engineering, RCoSE 2015*, pp. 12–18, 2015.

[3] R. Kohavi and R. Longbotham, "Online Controlled Experiments and A/B Tests," in *Encyclopedia of Machine Learning and Data Mining*, no. Ries 2011, 2015, pp. 1–11.

[4] H. H. Olsson and J. Bosch, *The HYPEX model: From opinions to data-driven software development*. 2014.

[5] E. Ries, *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. 2011.

[6] H. Li, W. Shang, and A. E. Hassan, "Which log level should developers choose for a new logging statement?," *Empirical Software Engineering*, vol. 22, no. 4, pp. 1684–1716, 2017.

[7] T. Barik, R. Deline, S. Drucker, and D. Fisher, "The Bones of the System: A Case Study of Logging and Telemetry at Microsoft," 2016.

[8] S. Marciuska, C. Gencel, and P. Abrahamsson, "Feature usage as a value indicator for decision making," *Proceedings of the Australian Software Engineering Conference, ASWEC*, pp. 124–131, 2014.

[9] H. Davenport, Thomas, "How to Design Smart Business Experiments," *Harvard Business Review*, vol. 0, 2009.

[10] R. Kohavi, A. Deng, B. Frasca, R. Longbotham, T. Walker, and Y. Xu, "Trustworthy online controlled experiments," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '12*, 2012, p. 786.

[11] M. Kim, T. Zimmermann, R. DeLine, and A. Begel, "The emerging role of data scientists on software development teams," in *Proceedings of the 38th International Conference on Software Engineering - ICSE '16*, 2016, no. MSR-TR-2015-30, pp. 96–107.

[12] S. D. Simon, "Is the randomized clinical trial the gold standard of research?," *Journal of Andrology*, vol. 22, no. 6, pp. 938–943, Nov. 2001.

[13] E. Bakshy, D. Eckles, and M. S. Bernstein, "Designing and deploying online field experiments," in *Proceedings of the 23rd international conference on World wide web - WWW '14*, 2014, pp. 283–292.

[14] R. Kohavi and S. Thomke, "The Surprising Power of Online Experiments," *Harvard Business Review*, no. October, 2017.

[15] A. Fabijan, P. Dmitriev, H. H. Olsson, and J. Bosch, "The Benefits of Controlled Experimentation at Scale," in *Proceedings of the 2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 2017, pp. 18–26.

[16] A. Deng, P. Zhang, S. Chen, D. W. Kim, and J. Lu, "Concise Summarization of Heterogeneous Treatment Effect Using Total Variation Regularized Regression," *In submission*, Oct. 2016.

[17] P. Dmitriev, S. Gupta, K. Dong Woo, and G. Vaz, "A Dirty Dozen: Twelve Common Metric Interpretation Pitfalls in Online Controlled Experiments," in *Proceedings of the 23rd ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '17*, 2017.

[18] D. Tang, A. Agarwal, D. O'Brien, and M. Meyer, "Overlapping experiment infrastructure," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '10*, 2010, p. 17.

[19] J. E. Guillory *et al.*, "Experimental evidence of massivescale emotional contagion through social networks," *Proceedings of the National Academy of Sciences*, vol. 111, no. 29, pp. 10779–10779, Jul. 2014.

[20] E. Lindgren and J. Münch, "Raising the odds of success: The current state of experimentation in product development," *Information and Software Technology*, vol. 77, pp. 80–91, 2015.

[21] S. Gupta, S. Bhardwaj, P. Dmitriev, U. Lucy, A. Fabijan, and P. Raff, "The Anatomy of a Large-Scale Online Experimentation Platform," in *to appear in Proceedings of the 2018 IEEE International Conference on Software Architecture (ICSA)*, 2018.

[22] J. F. Box, "R.A. Fisher and the Design of Experiments, 1922–1926," *The American Statistician*, vol. 34, no. 1, pp. 1–7, Feb. 1980.

[23] A. Fabijan, P. Dmitriev, H. H. Olsson, and J. Bosch, "The Evolution of Continuous Experimentation in Software Product Development: From Data to a Data-Driven Organization at Scale," in *Proceedings of the 2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, 2017, pp. 770–780.

[24] P. Dmitriev and X. Wu, "Measuring Metrics," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management - CIKM '16*, 2016, pp. 429–437.

[25] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical Software Engineering*, vol. 14, no. 2, pp. 131–164, 2008.

[26] R. S. Kaplan and D. P. Norton, "The Balanced Scorecard: Translating Strategy Into Action," *Harvard Business School Press*. pp. 1–311, 1996.

[27] A. Deng, J. Lu, and J. Litz, "Trustworthy Analysis of Online A/B Tests: Pitfalls, Challenges and Solutions," in *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, 2017, pp. 641–649.

[28] R. L. Kaufman, J. Pitchforth, and L. Vermeer, "Democratizing online controlled experiments at Booking. com," *arXiv preprint arXiv:1710.08217*, pp. 1–7, 2017.

[29] T. Crook, B. Frasca, R. Kohavi, and R. Longbotham, "Seven pitfalls to avoid when running controlled experiments on the web," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '09*, 2009, p. 1105.

[30] Z. Zhao, M. Chen, D. Matheson, and M. Stone, "Online Experimentation Diagnosis and Troubleshooting Beyond AA Validation," in *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 2016, no. October 2016, pp. 498–507.

[31] H. Hohnhold, D. O'Brien, and D. Tang, "Focusing on the Long-term," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '15*, 2015, pp. 1849–1858.